Engineering and Computational Mechanics

A tablet computer application for conceptual design
Åkesson and Lindemann

ice | proceedings

Proceedings of the Institution of Civil Engineers

http://dx.doi.org/10.1680/eacm.14.00020
Paper 1400020
Received 18/08/2014 Accepted 05/03/2015
Keywords: computational mechanics/design methods & aids/education & training

ice
Institution of Civil Engineers
publishing

A tablet computer application for conceptual design

Daniel Åkesson MSc
PhD student, Lund University, Lund, Sweden

Jonas Lindemann PhD
Researcher, Lund University, Lund, Sweden

In the conceptual design phase, solutions are reached through an iterative, high-paced and often chaotic manner. Conventional advanced structural analysis software is often too advanced and insufficiently agile to follow this high-paced work pattern. Premature use of advanced structural analysis tools can negatively affect conceptual understanding and the quality of the conceptual design. The multi-touch interfaces of today's tablet computers give the user a strong feeling of direct manipulation of objects on the screen. This is interesting for structural mechanics applications, enabling direct manipulation of the structural model on the screen so as to have a better understanding and feeling of the structural behaviour. A tablet computer application for the conceptual design phase, which uses this type of direct manipulation interface, has been developed.

1. Introduction

> In practice a great many problems are solved by what is called judgment. The better a man understand how the stresses follow through a member or structure, the better his judgment will be. (Wolfe, 1921)

The earliest phase of the design process is referred to as the conceptual design stage. This stage is distinguished from the later design stages by more general questions and solutions (McNeill *et al*., 1998). The conceptual design stage starts with a general description of a problem and ends, after the creation and exploration of new ideas, with a general description of a solution (McNeill *et al*., 1998), in this case a structure. A structure is born during the conceptual design phase, and problems that arise in later design stages are often a result of careless conceptual design (Schlaich, 2006).

In the conceptual design phase, solutions are reached through an iterative, high-paced and often chaotic manner (Schlaich, 2006). Conventional advanced structural analysis software is often too advanced and insufficiently agile to follow such highly paced work patterns. It has been shown that premature use of advanced structural analysis tools can negatively affect the conceptual understanding and the quality of the conceptual design (Fröderberg and Crocetti, 2014).

The importance of the conceptual design stage is often overlooked and structural aspects are only considered in a late design stage (Schlaich, 2006). Structural engineers are in need of an improved conceptual design toolbox (Fröderberg and Crocetti, 2014), and new computer-aided tools could be a part of this toolbox.

New technology opens up new possibilities for computer-aided tools in the conceptual design phase. The iPhone, launched in 2007 (Kerris and Dowling, 2007), featured a revolutionary new input technology – the multi-touch interface. This is a touch screen that allows interaction with multiple fingers simultaneously, making it possible to interact using gestures, such as pinch to zoom or swipe to turn page.

The multi-touch interface has closed the gap between human and computer, giving the user a stronger feeling of directly manipulating objects on a screen (Sears *et al*., 1993). This makes it interesting for conceptual structural design applications, giving the user the ability to directly manipulate the model on the screen, which could result in a better understanding and feeling of structural behaviour. These possibilities are explored in this paper. A measurement of normalised redundancy has also been implemented; for further details see Tibert and Achi (2012).

2. Previous work

The emerging field of conceptual structural design computation seeks to close the gap between visualisation tools and computational analysis tools (Mueller, 2014). Numerous applications exist for conceptual structural design. Two different key features – guidance and feedback – can be identified for design tools that encourage integrated conceptual design (Mueller, 2014).

2.1 Guidance features

Applications with guidance features suggest new geometries to the user in order to improve the structural performance of the model (Mueller, 2014). These applications make use of different optimisation techniques to compute new geometries that are presented for the user. Two interesting similar applications that make use of topology optimisation are Forcepad (Lindemann *et al*., 2004) and Topopt (Aage *et al*., 2013); in these two applications, a two-dimensional geometry is modelled, a topology optimisation is then performed and the resulting optimised shape is visualised.

Another optimisation technique that can be used for conceptual structural design is the genetic algorithm (Goldberg, 1989). The two applications Structurefit (Mueller, 2014) and Evolutionary design tool (Von Buelow, 2008) make use of this technique to optimise the shape of truss networks. One of the strengths of using evolutionary computing in this context is that local optima can be determined and presented to the user as structurally well-performing design alternatives.

2.2 Feedback features

This type of application responds quickly to the user's input, ideally in real-time, to allow for an interactive user experience (Mueller, 2014). Numerous applications that implement real-time numerical analysis have been developed both for practice and for research. Two of the first applications to implement this approach were Pointsketch (Olsson, 2006), which was the inspiration for the present work, and an application called Arcade (Martini, 2006). These two applications make use of the finite-element method and bar elements, which the user can interact with using a mouse and keyboard user interface.

The commercial and widespread finite-element analysis (FEA) software SAP2000 introduced, in version 12, a 'model alive' feature that gives real-time feedback for forces and deformations in truss networks (Clune *et al*., 2012). More recently, Autodesk – a software company known for visualisation tools – launched its new application ForceEffect both as a tablet and as a web application (Autodesk, 2015). The application is a tool for engineers to visualise and analyse truss networks, and also has support for visualising rigid body motions. According to Mueller (2014), 'the advantage of this class of tools over traditional structural analysis programs is the speed with which they convey results'.

3. FEA software in the conceptual design phase

Conventional FEA software is designed for later design stages, not the conceptual design phase. As the conceptual design phase has more general questions and solutions (McNeill *et al*., 1998), software needs to be adapted accordingly. Often, conventional software is too complicated and time consuming to use, which does not integrate well with a designer's iterative workflow (Lindemann *et al*., 2010).

4. Sketch-a-frame – a conceptual structural design tool

Sketch-a-frame is a finite-element iPad (Apple, 2014a) application for the conceptual design phase that makes use of beam elements. The application is free to download and available in the Appstore for iPad. The user interface is designed with the intention to be fast, intuitive and easy to understand. Instead of the conventional simulation sequence (see Figure 1), a more direct manipulation cycle is used, as shown in Figure 2.

The result is visualised when the model is considered complete; that is, if boundary conditions and one or more forces have been applied. When the model is complete it is determined if any rigid body motions are possible by calculating the determinant of the stiffness matrix; if the determinant is non-zero the model is stable. If the model is not stable and rigid body motions are possible an eigenvalue analysis is performed; the first modal shape is then visualised with an animation.

Computations are performed in real-time as changes are made to the geometry (e.g. when a node is moved the result is continuously updated and visualised; see Figure 2). Real-time visualisation encourages the user to experiment with the model in an explorative manner.

The application presents no numerical values on the geometry, forces or deformations. This is a design decision to encourage the user to focus on the general structural behaviour and not on the exact numerical results. This is not a tool to dimension structural members and therefore the material properties used are not relevant.

5. Theory and implementation

The application was developed using C++ and Objective-C. Finite-element computations are performed in C++ for performance reasons, using an external library called Newmat (Davies, 2014) for matrix operations. Newmat contains classes for common matrix and vector operations. The user interface was developed using Objective-C.

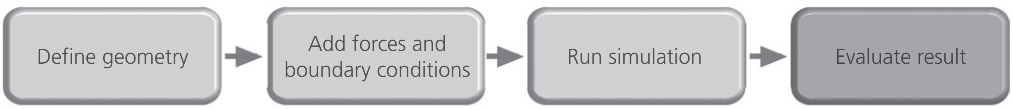

Figure 1. Conventional simulation sequence

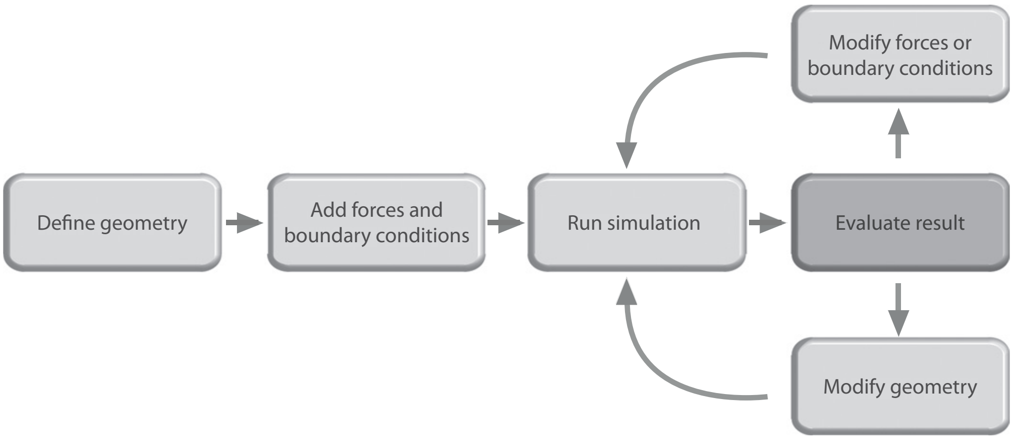

Figure 2. Direct manipulation cycle

5.1 Finite-element model

Beam elements (Bathe, 1996) are used for the computations. Every node has three degrees of freedom (DOFs) – vertical, horizontal and rotational – as shown in Figure 3.

5.2 Eigenvalue analysis

From the stiffness matrix $\mathbf{K}$ of a structure, a set of scalar stiffness values can be determined (Olsson and Thelin, 2003). Assume that a set of displacements $\boldsymbol{a}$ exists, which are proportional to a corresponding set of forces $\boldsymbol{f}$

$$\boldsymbol{f} = \lambda \boldsymbol{a}$$

This can be combined with a linear elastic finite-element formulation

$$\mathbf{K}\boldsymbol{a} = \boldsymbol{f} = \lambda \boldsymbol{a}$$

which can be rewritten as

$$(\mathbf{K} - \lambda \mathbf{I})\boldsymbol{a} = 0$$

This is a standard eigenproblem. The eigenvalues λ have dimensions of force/length, also called canonical stiffness values (Olsson, 2006). Every eigenvalue λ_i has a corresponding eigenvector $\boldsymbol{a}_i$, which describes a modal shape. Eigenvalues equal to zero means zero energy is required to form the corresponding modal shape (i.e. a rigid body motion). The eigenvectors are only defined within a scalar multiple. The eigenvector is normalised and multiplied by a positive and negative scalar, and the result is two different shapes. An animation of the rigid body motion can be achieved by interpolating between the two different shapes; this is implemented in Sketch-a-frame.

Figure 3. Beam element with six DOFs

5.3 Static redundancy factor

In Sketch-a-frame, normalised static redundancy factors are presented as a way to assess the redundancy of individual elements in the structure (Tibert and Achi, 2012). Only topology and geometry are studied in the analysis. Using the equilibrium matrix $\mathbf{H}$, described in detail by Pellegrino and Calladine (1986), and the diagonal elements stiffness matrix $\boldsymbol{\Psi}$ yields (Tibert and Achi, 2012)

$$\boldsymbol{\Lambda} = \mathbf{I} - \mathbf{H}^{\mathrm{T}}(\mathbf{H}\Psi\mathbf{H}^{\mathrm{T}})^{-1}\mathbf{H}\Psi$$

The trace of the scaling matrix $\boldsymbol{\Lambda}$ is equal to the degree of static indeterminacy s

$$\mathrm{tr}(\boldsymbol{\Lambda}) = s$$

Thus, every diagonal element Λ_{ii} can be interpreted as element i's contribution to the structure's static indeterminacy, and is hence denoted the static redundancy factor. In Sketch-a-frame, the result is normalised and visualised using a colour scale (see Figure 4). Elements coloured blue cannot be removed without rendering the structure unstable, while elements coloured red are highly redundant. If there are any changes in the geometry (e.g. a bar is added or removed), the updated result is computed and visualised automatically.

In Example A of Figure 5, the vertical bar is 0% redundant as it is the only bar that provides vertical stiffness to the structure.

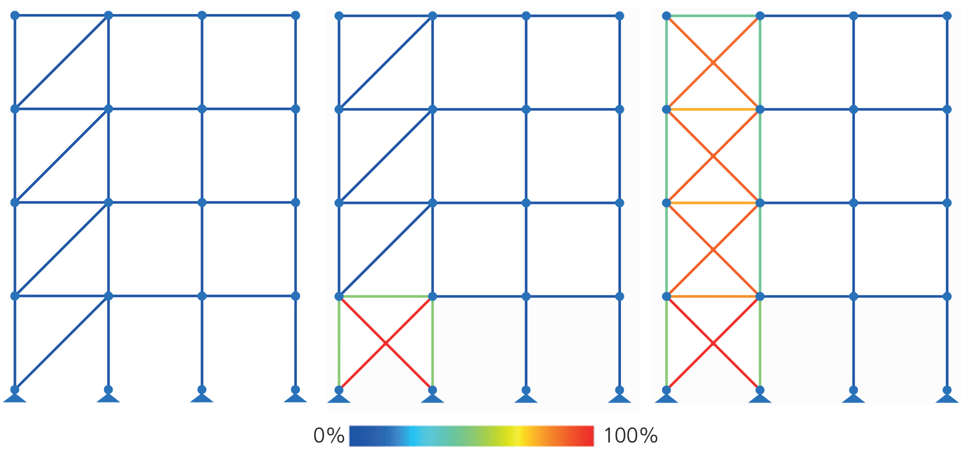

Figure 4. Normalised degree of redundancy

In Example B, the element coloured red is the most redundant element, but any element could be removed without rendering the structure unstable.

6. User interface

To make it easy for the tablet user to understand how different applications work it is important that there is consistency between them. Apple has guidelines for developers (Apple, 2014b). How an application works should not be based on the capabilities of the device, but on the way people think and work. The application should also be optimised for the device that it is running on. Aesthetic integrity is also of importance; it is not only about creating something beautiful, it is also how the design integrates with the functionality of the application.

Users enjoy direct manipulation on the screen – it keeps them engaged and gives a feeling of control (Apple, 2014b). Direct manipulation is a user interface style with continuous representation of objects of interest with rapid, reversible and incremental feedback (Shneiderman, 1982). Users can directly manipulate objects on the screen using real-world metaphors.

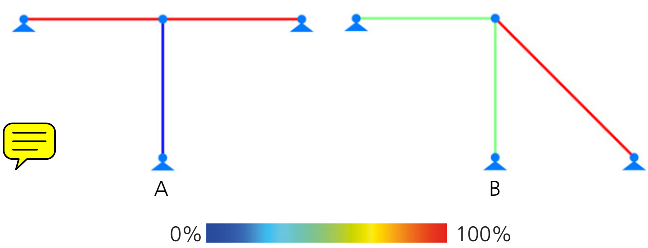

Figure 5. Normalised degree of redundancy, further examples

This user interface style creates intuitive user interfaces that work very well with the multi-touch screen (Sears *et al.*, 1993).

6.1 General layout of application

During the development process of the user interface, the quick and dirty evaluation paradigm (Rogers *et al.*, 2011) was used to quickly get feedback from users. In this paradigm, users are observed as they interact with the application with minimal control from the evaluators. Later in the development process, the usability testing evaluation paradigm (Rogers *et al.*, 2011) was used to review the performance of the interface and find further improvements. In the evaluation paradigm, users were given a task to perform as they were recorded using a camera, to be used for later assessments.

The goal of the user interface development was to create a simple to use and intuitive user interface. Feedback from the users guided the development to a user interface with three different pre-set visualisation modes (see Figure 6) available in a tab bar. The different modelling tools are always visible and accessible for the user in a toolbar. The three different visualisation modes are

- deformations
- normal force and moment
- normalised redundancy.

Figure 6. The three different visualisation modes

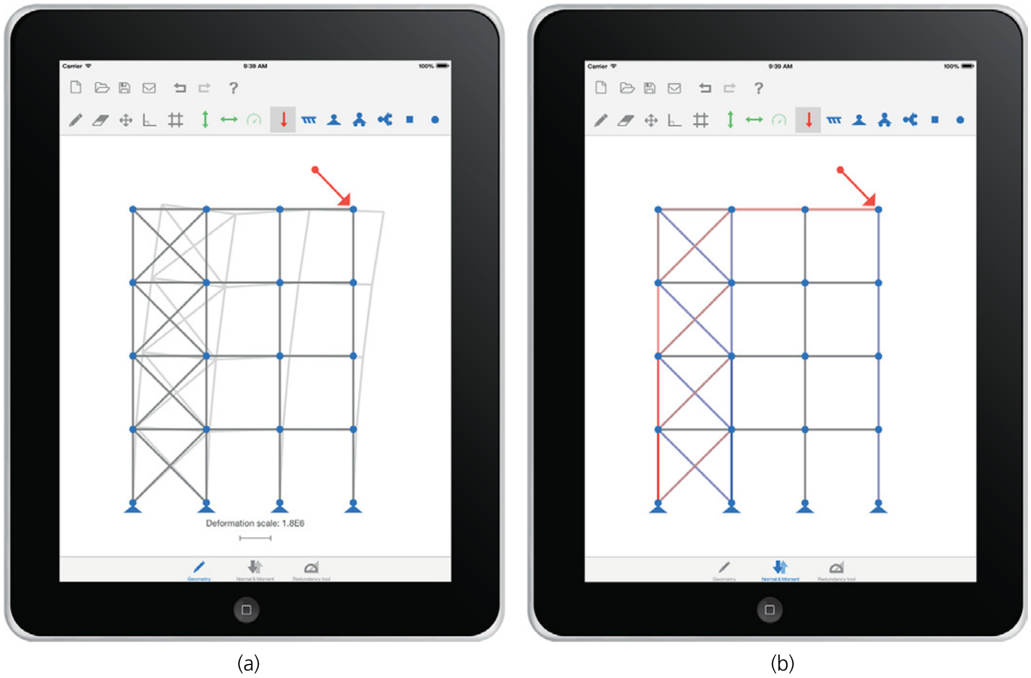

Figure 7. (a) Deformations visualised; (b) normal force visualised

In the deformations mode, the deformed shape is visualised (see Figure 7(a)). In the normal force and moment mode, the normal force is visualised by colouring the elements red (for tension) and blue (for compression), as shown in Figure 7(b). If moment constraints exist, a moment envelope is visualised.

The application also has support for sharing the result. A pre-composed e-mail can be sent containing all the results in image format and a model file, which can be re-opened in the application.

6.2 Geometric modelling

As shown in Figure 8, the modelling tools available to the user in the toolbar are

- pen – swipe to create elements and nodes
- eraser – swipe or tap to remove elements and nodes
- move – moves nodes; result is visualised in real-time during the movement
- undo and redo – changes can be undone and redone
- grid – a grid is shown that the tools snap against
- ortho – elements created are vertical or horizontal.

While using the pen tool and a swipe gesture is initiated, a node is created; when the gesture is released, an end node and an element between the nodes is created, as shown in Figure 9(a). When a swipe gesture is active, an element is visualised to the current position; if the current position is close to a node or a

Figure 8. Modelling tools

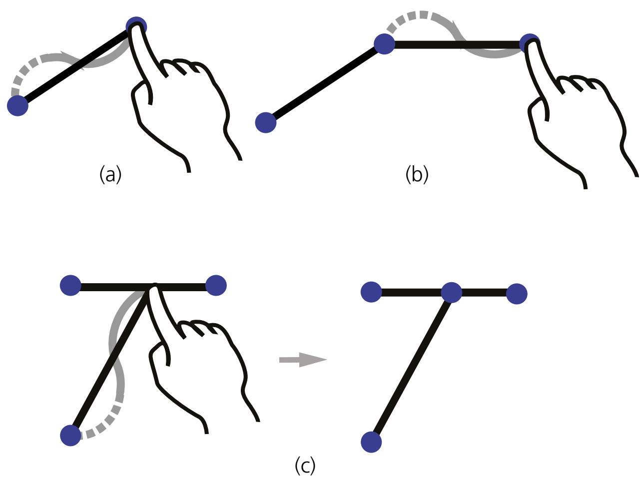

Figure 9. (a) Element created on release; (b) element connected from existing node; (c) new element connected to existing on release

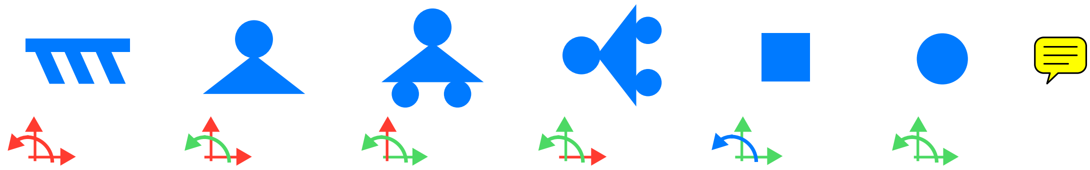

Figure 10. Boundary conditions: red arrows represent constrained DOFs, green arrows represent unconstrained DOFs and blue arrow can transfer bending moment between elements

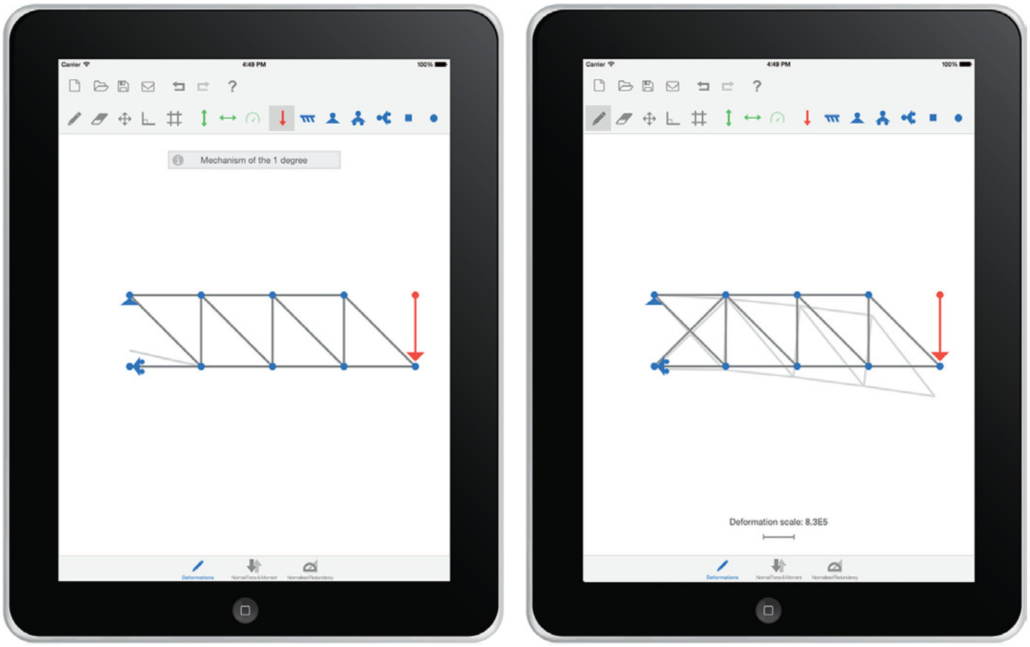

Figure 11. Assignment solved

line, the visualised element snaps to it. This also fits well in the direct manipulation of objects with continuous representation of objects of interest. This allows for a line or node to be the start or the end point of a new element (see Figures 9(b) and 9(c)).

6.3 Boundary conditions and forces

As beam elements are used, every node has three DOFs (i.e. vertical and horizontal translation and rotation), which can be constrained in different combinations. For simplicity, six of the most common combinations are implemented with symbols familiar to engineers (see Figure 10).

A force tool is also available. Tapping or swiping on nodes using this tool adds forces to the selected node. Forces can also be adjusted using the force tool and as the force is adjusted the result is visualised in real-time.

6.4 Scale

The difference in deformations between two models can be up to a factor of 10^9 times, depending on the stiffness of the model. To be able to visualise the deformation independently of the stiffness of the models, it needs to be scaled. However, constant rescaling removes some of the important feeling of directly manipulating the model. Rescaling when a swipe gesture is released was the best solution found; this retains the feeling of direct manipulation and allows the deformations to be visualised properly.

The current scale is presented to the user at the bottom of the screen together with a scale bar. The scale bar is always the same length as the largest deformation, and rescales when a swipe gesture is released. The same concept is used for visualising the moment envelope.

7. Application usage examples

7.1 Development of a roof truss structure in glulam

The application was used by two civil engineering students in their master thesis work to improve the market competiveness of glulam by developing a standardised concept for roof trusses for large open-space structures (Hedlund and Brorson, 2013). The following work process was used to find the most efficient truss designs.

- Inspiration and brainstorming – a large number of constructions was studied and evaluated. Inspiration was found from existing structures and literature, with the most successful constructions moving on to the next stage.
- Sketching – constructions were refined and improved.
- Rough design – cross-sectional forces were calculated using the software SAP2000 (CSI, 2015).
- Cost estimate based on cross-section and number of nodes.
- Selection based on comparison of stiffness, cost, number of nodes and installation.
- Final design, optimisation and cost estimation – roof trusses from the selection were further refined and a new cost estimation performed before the final design was chosen.

Sketch-a-frame was used in the first two stages in which the students brainstormed and explored different designs. The students preferred using the application over more advanced software in this conceptual design phase as a result of the speed of the interaction. The students also reported an increased initial understanding of the structural behaviour of the models.

7.2 Solving a course assignment

Some civil engineering students were given an assignment to solve. The task was to analyse a structure and determine if it was stable and, if not, give examples of improvements to make it stable. Most students solved the assignment by using Matlab and calculating the determinant of the stiffness matrix, but one of the students used Sketch-a-frame instead; the structure was quickly modelled and the result was immediately presented (see Figure 11). With animation of the rigid body motion, this student understood how the structure could be improved to make it stable and could quickly try different solutions.

8. Conclusions

The strength of tools with feedback features lies in the speed of the interaction and the fact that a freer exploration of structural forms is allowed. Sketch-a-frame further improves on the strengths of feedback feature tools as the speed of the interaction is improved with the multi-touch interface.

The multi-touch interface also gives a stronger feeling of directly manipulating the model on the screen, which encourages exploration and gives the user a feeling of control. With a higher speed of interaction, structural forms can be explored more freely compared with earlier mouse and keyboard feedback tools. However, some precision is lost with the multi-touch interface that can be a disadvantage for the geometric modelling of details. Another advantage of a tablet application is that it can easily be brought to meetings and used as a discussion tool, which can create a collaborative environment between participants.

The process of conceptual and structural design can be described by four steps – conceiving, modelling, dimensioning and detailing (Schlaich, 2006). In engineering, however, there is a lack of tools that support the first steps of conceiving and modelling. Sketch-a-frame provides such a tool. The development of the glulam roof structure is a good example of how Sketch-a-frame can be used to improve the conceptual and structural design process.

REFERENCES

Aage N, Nobel-Jørgensen M, Andreasen CS and Sigmund O (2013) Interactive topology optimisation on hand-held devices. *Structural and Multidisciplinary Optimisation* **47(1)**: 1–6.

Apple (2014a) *iPad*. Apple, San Fransisco, CA, USA. See http://www.apple.com/ipad/ (accessed 25/03/2015).

Apple (2014b) *iOS Human Interface Guidelines*. Apple, San Fransisco, CA, USA. See http://developer.apple.com/library/ios/documentation/userexperience/conceptual/MobileHIG/index.html (accessed 25/03/2015).

Autodesk (2015) *ForceEffect*. See http://www.autodesk.com/forceeffect (accessed 07/04/2015).

Bathe KJ (1996) *Finite Element Procedures*. Prentice-Hall, Englewood Cliffs, NJ, USA.

Clune R, Connor JJ, Ochsendorf JA and Kelliher D (2012) An object-oriented architecture for extensible structural design software. *Computers and Structures* **100**: 1–17.

CSI (Computers and Structures, Inc.) (2015) *SAP2000*. CSI, Walnut Creek, CA, USA. See www.csiamerica.com/sap2000 (accessed 25/03/2015).

Davies R (2014) Newmat C++ matrix library. See http://www.robertnz.net/index.html (accessed 25/03/2015).

Fröderberg M and Crocetti R (2014) Engineers in need of an improved conceptual design toolbox. *Proceedings of 37th IABSE Symposium, Madrid, Spain*, vol. **102**, pp. 515–521.

Goldberg DE (1989) *Genetic Algorithms in Search, Optimisation, and Machine Learning*. Addison-Wesley, Boston, MA, USA.

Hedlund R and Brorson A (2013) *Development of a Roof Structure in Glulam*. Master's dissertation, Lund University of Technology, Lund, Sweden.

Kerris N and Dowling S (2007) *Apple reinvents the phone with iPhone*. Macworld, San Francisco, CA, USA.

Lindemann J, Sandberg G and Olsson KG (2004) An approach to teaching architectural and engineering students utilizing computational mechanics software ForcePAD. *Journal of Information Technology in Construction* **9**: 219–228.

Lindemann J, Sandberg G and Damkilde L (2010) Finite-element software for conceptual design. *Proceedings of the Institution of Civil Engineers – Engineering and Computational Mechanics* **163(1)**: 15–22, http://dx.doi.org/10.1680/eacm.2010.163.1.15.

Martini K (2006) A new kind of software for teaching structural behavior and design. In *Proceedings of the 2006 Building Technology Educators Conference, College Park, MD, USA*.

McNeill T, Gero JS and Warren J (1998) Understanding conceptual electronic design using protocol analysis. *Research in Engineering Design* **10(3)**: 129–140.

Mueller CT (2014) *Computational Exploration of the Structural Design Space*. PhD thesis, Massachusetts Institute of Technology, Boston, MA, USA.

Olsson KG and Thelin C (2003) *Use of Static Eigenmodes in Mechanical Design*. Chalmers University of Technology, Gothenburg, Sweden.

Olsson P (2006) *Conceptual Studies in Structural Design: Pointsketch – A Computer Based Approach for use in Early Stages of the Architectural Process*. Unpublished doctoral thesis, Chalmers University of Technology, Gothenburg, Sweden.

Pellegrino S and Calladine CR (1986) Matrix analysis of statically and kinematically indeterminate frameworks. *International Journal of Solids and Structures* **22(4)**: 409–428.

Rogers Y, Sharp H and Preece J (2011) *Interaction Design: Beyond Human–Computer Interaction*. Wiley, Chichester, UK.

Schlaich M (2006) Challenges in education – conceptual and structural design. *Proceedings of 31st IABSE Symposium, Budapest, Hungary*, vol. 92, pp. 20–26.

Sears A, Plaisant C and Shneiderman B (1993) A new era for high precision touchscreens. In *Advances in human-computer interaction* (Hartson HR and Hix D (eds)). Ablex Publishing Corp., Norwood, NJ, USA, vol. 3, pp. 1–33.

Shneiderman B (1982) The future of interactive systems and the emergence of direct manipulation. *Behaviour & Information Technology* **1(3)**: 237–256.

Tibert G and Achi LM (2012) Static redundancy factors in conceptual design. In *Proceedings of 25th Nordic Seminar on Computational Mechanics, Lund, Sweden* (Persson K, Revstedt J, Sandberg G and Wallin M (eds)), pp. 261–267.

Von Buelow P (2008) Suitability of genetic based exploration in the creative design process. *Digital Creativity* **19(1)**: 51–61.

Wolfe WS (1921) *Graphical Analysis: A text book on graphic statics*. McGraw-Hill Book Company, New York, NY, USA.